WHAT IS CLAIMED IS:

1. An edge counter comprising:

an input receiving an input signal and an output on which an output signal is driven; and

a set of logic gates between the input and output, the logic gates configured to change a state of the edge counter with each transition of the input signal and to produce an output signal having a cycle corresponding to a predetermined number of transitions of the input signal.

2. The edge counter according to claim 1, wherein the predetermined number may be odd or even.

3. The edge counter according to claim 1, wherein a signal path between the input and output through the logic gates includes a sequence of only two logic gates.

4. The edge counter according to claim 1, wherein the logic gates generate a set of intermediate signals, at least one of the intermediate signals changing state in response to transition of the input signal.

5. A wireless receiver including the edge counter according to claim 1, the wireless receiver further comprising one of a local oscillator and a clock divider employing the edge counter.

6. A wireless communications system including the wireless receiver according to claim 5, the wireless communications system further a wireless transmitter and a communications path between the transmitter and the receiver.

7. A method of designing an edge counter comprising:

defining a number of intermediate signals sufficient to count a predetermined number of edges;

determining states of the intermediate signals to be generated; and

from the determined states, deriving a set of logic gates receiving an input signal, generating the intermediate states in response to transitions in the input signal, and producing an output signal having a cycle corresponding to the predetermined number of edges within the input signal.

8. The method according to claim 7, further comprising:

inserting gray codes for states of the intermediate signals in a table in a manner corresponding to changes based on input clock signal transitions.

9. The method according to claim 8, further comprising:

inserting the gray codes in the table to correspond to a transition in the output signal.

10. The method according to claim 9, further comprising:

identifying rows containing gray codes matching a row value.

11. The method according to claim 10, further comprising:

generating a Karnaugh map for the states of the intermediate signals corresponding to the identified rows; and

designing a set of logic gates to implement the logic function represented by the Karnaugh map.

12. The method according to claim 11, further comprising:

generating a Karnaugh map for each of the intermediate signals and the output signal.

13. The method according to claim 7, further comprising:

designing the logic gates to have a two gate delay between the input signal and the output signal.

14. An edge counter designed by the steps of:

defining a number of intermediate signals sufficient to count a predetermined number of edges;

determining states of the intermediate signals to be generated; and

from the determined states, deriving a set of logic gates receiving an input signal, generating the intermediate states in response to transitions in the input signal, and producing an output signal having a cycle corresponding to the predetermined number of edges within the input signal.

15. The edge counter according to claim 14, further designed by the step of:

inserting gray codes for states of the intermediate signals in a table in a manner corresponding to changes based on input clock signal transitions.

16. The edge counter according to claim 15, further designed by the step of:

inserting the gray codes in the table to correspond to a transition in the output signal.

17. The edge counter according to claim 16, further designed by the step of:

identifying rows containing gray codes matching a row value.

18. The edge counter according to claim 17, further designed by the steps of:

generating a Karnaugh map for the states of the intermediate signals corresponding to the identified rows; and

designing a set of logic gates to implement the logic function represented by the Karnaugh map.

19. The edge counter according to claim 18, further designed by the step of:

generating a Karnaugh map for each of the intermediate signals and the output signal.

20. The edge counter according to claim 14, further designed by the step of:

designing the logic gates to have a two gate delay between the input signal and the output signal.